

## E se criássemos um jogo em Scratch?

**João Torres**  
jvtorres@gmail.com

**Cristina Carrilho**  
crisn2006@gmail.com

### Introdução

Scratch é uma linguagem de programação concebida no Massachusetts Institute of Technology (MIT) sob o lema “imagina, programa, partilha”. É uma ferramenta concebida para apoiar o desenvolvimento da fluência tecnológica em crianças, desde muito cedo, permitindo assim que estas adquiram as competências transversais ditas do “para o séc. XXI”. É uma linguagem de programação extremamente acessível por não exigir conhecimentos prévios de outras linguagens e não exigir a escrita de comandos. Está traduzida em muitas línguas e tem capacidades multimédia permitindo a inserção de imagens e sons.

Este documento não pretende ser um guia exaustivo do programa. Tem por objetivo fornecer uma primeira abordagem a iniciantes que tencione criar um jogo em Scratch.

### O ambiente de programação

Comece por aceder, no seu *browser*, a <http://scratch.mit.edu/>



Figura 1. Página inicial <http://scratch.mit.edu>

O site reconhece que estamos a aceder a partir de Portugal e, por essa razo, a pgina  apresentada em portugus. Ser neste site que poderemos criar programas em Scratch e ainda explorar, partilhar e comentar programas criados por scratchers de todo o mundo.<sup>78</sup>

No cimo do ecr,  esquerda, esto em destaque trs grandes opes que exibimos na figura 2.



Figura 2. Trs opes

Vamos comear por seleccionar a primeira destas opes para darmos incio  nossa aventura. A segunda opo permitiria ver exemplos de programas feitos em Scratch e a terceira registarmo-nos gratuitamente para, posteriormente, partilhar o nosso trabalho com a comunidade. Este aspeto de partilha  muito importante, pelo que nos registaremos mais tarde. Ser nesta pgina, diretamente no nosso browser, representada na figura 3, que programaremos o nosso jogo.

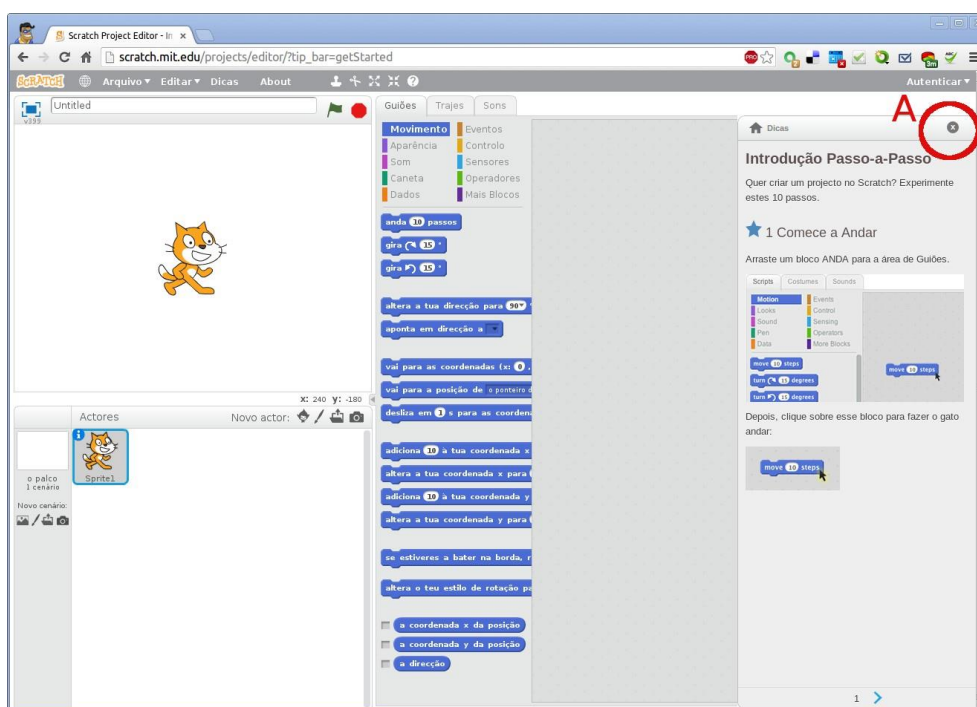


Figura 3. Ambiente de programao Scratch

Este ecr est dividido em vrias zonas que comearemos a explorar de seguida. Do lado direito abre-se uma aba que nos permite dar os primeiros passos em Scratch e que, se estiver com tempo, aconselhamos a seguir. Para continuar este roteiro podemos fechar essa aba, fazendo

<sup>78</sup> Caso queira utilizar o scratch sem ligao  Internet pode descarregar uma verso que funciona sem estar ligado  rede aqui: <http://scratch.mit.edu/scratch2download/>

clique no botão assinalado (fig. 3 A) para ficarmos com mais espaço  til nesta fase.

Se repararmos no ecr  podemos distinguir 4  reas que destacamos na figura 4.

## O jogo do labirinto

### Escolher uma personagem

O nosso objetivo   criar um labirinto que ter  como protagonista um simp tico fantasma<sup>79</sup>. Vamos portanto “dispensar” o nosso gato fazendo clique sobre ele, com o bot o do lado direito do rato, (fig. 4 B) e escolhendo depois a op o “remover”

O passo seguinte   criar a nossa personagem: o fantasma! Existem 4 modos de criar novas personagens, destacadas na figura 6:

Figura 6 -A Escolher um ator da biblioteca do Scratch.

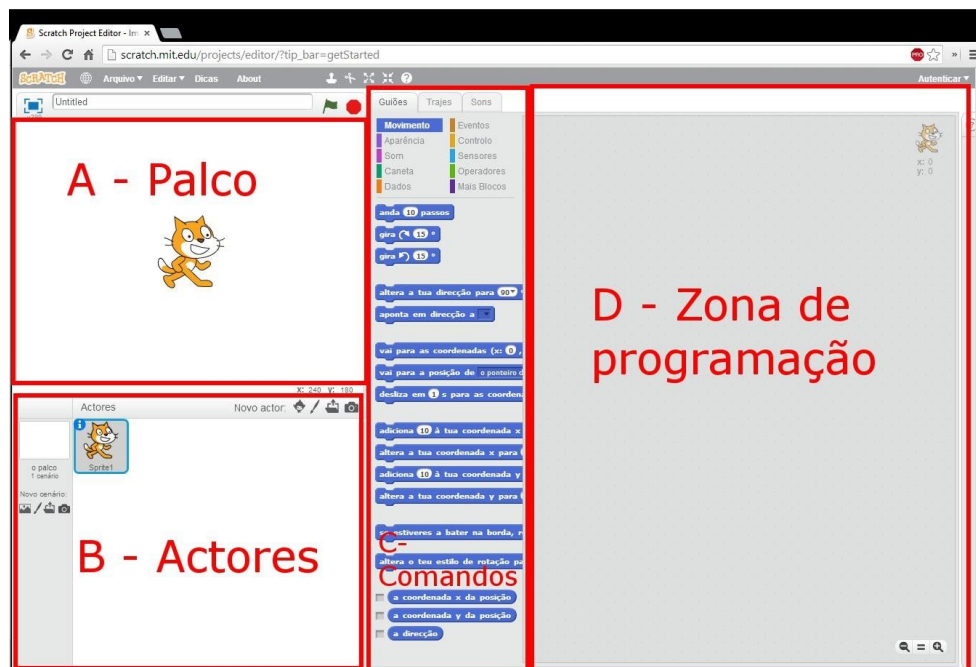


Figura 4. Principais  reas do ambiente de programac o

Figura 6 -B Desenhar um ator no editor interno do Scratch

Figura 6 -C Carregar um ator a partir de uma imagem gravada no nosso computador

Figura 6 -D Tirar uma foto com a c mara do PC e us -la como ator.

<sup>79</sup> Pode ver aqui <http://scratch.mit.edu/projects/19519597/> o resultado final pretendido



Figura 5: Remover uma personagem

Figura 6: Criar um novo ator

Escolhemos a primeira das opções (fig. 6 A) e teremos acesso à biblioteca de personagens do Scratch representada na figura 7.

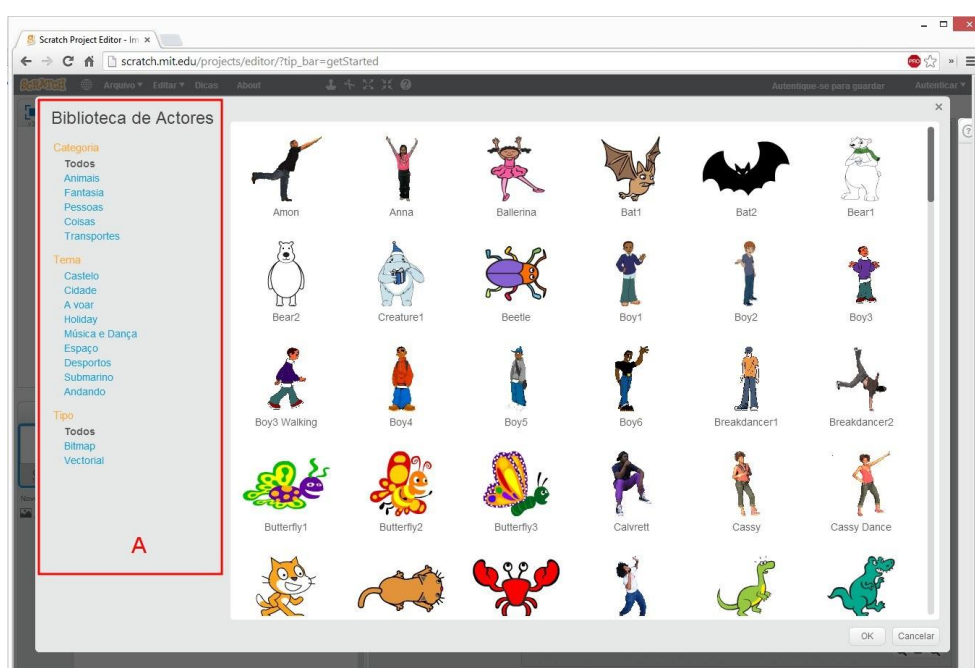


Figura 7. Escolher um ator

Podemos filtrar essas imagens por categoria, tema ou tipo (fig. 7 A). Assim escolhendo “fantasia” encontraremos facilmente o nosso herói, o fantasma! Pode, no entanto, escolher qualquer outro.

### Movimentar o fantasma

O nosso fantasma está agora no centro do ecrã. Vamos programá-lo para que se movimente quando premir as setas do cursor.

A programação em Scratch é feita arrastando os comandos que se encontram na zona de comandos (fig. 4 C) para a zona de programação (fig. 4 D). Os comandos disponíveis estão agrupados por categoria, havendo ainda um código de cores que facilita a sua identificação. Assim, os comandos de movimento, por exemplo, são todos azuis e aparecem quando clicamos no botão “movimento” (fig. 8 A).

Vamos arrastar o comando “Anda (10) passos” da zona de comandos para a zona de programação. Este comando, sempre que executado, faz “andar” o nosso personagem 10 passos. Podemos executar o comando (ou um bloco de comandos) fazendo duplo clique sobre ele. Quando fazemos duplo clique reparamos que o fantasma se desloca, 10 passos, da esquerda para a direita.

O facto de se deslocar para a direita é mero acaso, nada do que disse até agora garante que assim acontea. Por omisso, quando criamos um novo ator, ele est orientado para a direita e, por isso,  nessa direo que se desloca.

No caso de querer garantir que ser nesta direo que a deslocao acontece temos que juntar antes do comando de movimento outro que o “obrigue” a, antes de andar, alterar a direo para a direita (90o).

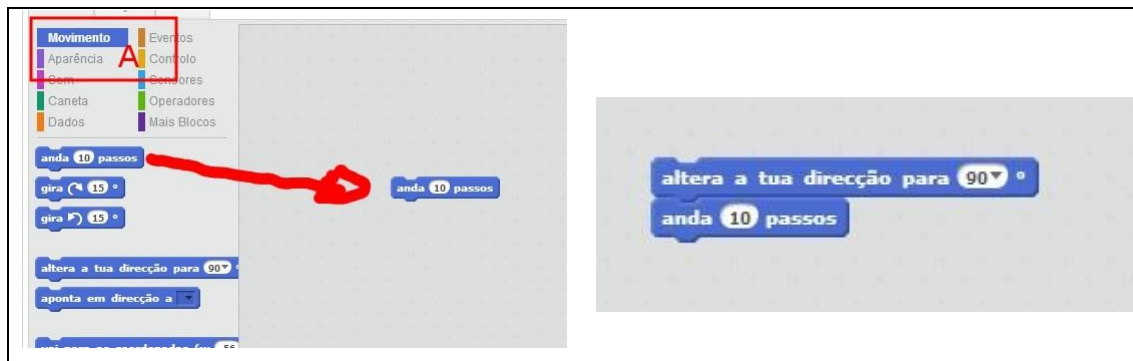


Figura 8. Mover o fantasma

Figura 9. Virar para a direita e mover

Reparem que os comandos esto concebidos para encaixarem uns nos outros. Notem tambm que a ordem  importante, para andar para a direita  necessrio primeiro garantir que o fantasma est orientado nessa direo e apenas depois se deve movimentar.

Se fizermos agora duplo clique sobre o primeiro comando sero ambos executados e, agora, temos a certeza que o fantasma se desloca na direo certa!

Para garantir que estes comandos so executados quando premirmos a tecla “seta para a direita” juntamos o comando “Quando algum pressionar a tecla” e escolhemos a tecla “seta para a direita” (fig. 10)

Este comando est no grupo dos “Eventos” e, se reparar, no  possvel encaixar nenhum outro comando antes, apenas depois.

Assim, definimos que “quando algum o pressionar a tecla seta para a direita o fantasma vira para a direita e move 10 passos”.

Tente!

Quando carregarmos na seta para a esquerda algo de muito parecido acontecer. Neste caso apenas a direo ser outra. Se fizermos clique sobre o primeiro dos comando do bloco aparece no menu contextual a opo duplicar. Assim teremos outro bloco de comandos e bastar fazer as alteraes (fig. 11).

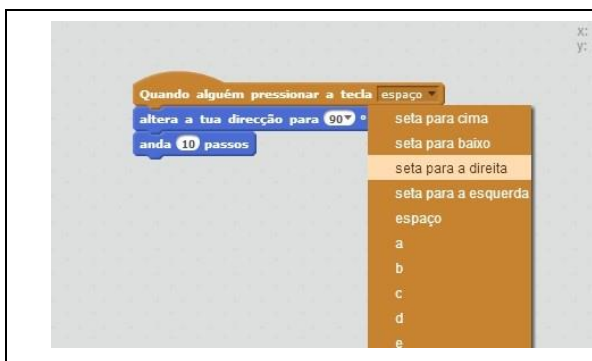


Figura 10. Quando alguém pressionar a tecla...



Figura 11. Duplicar comandos

Pelo mesmo processo crie os comandos para as setas para cima e para baixo que permitirão mover o fantasma nestas direões (fig. 12). Neste momento, o nosso fantasma movimenta-se nas quatro direões de acordo com os comandos que damos via teclado, utilizando as teclas direcionais.

Repare, no entanto, que o fantasma roda livremente ficando por vezes invertido. Existe uma propriedade que define o modo como os atores giram sobre si mesmos.



Figura 12. Comandos para movimentar nas quatro direões

Para acedermos a essa e outras propriedades fazemos clique ícone “i” azul, destacado na Figura 13. Devem ser mostradas as propriedades do nosso ator (fig. 14) e, entre elas, temos a possibilidade de alterar o estilo de rotaão. Existem 3 estilos disponíveis, altere e faça testes de modo a perceber o efeito de cada um destes três estilos diferentes.

Podemos também alterar o nome da nossa personagem, eu optei por chamar “fantasma” e optar pelo modo em que pode apenas girar horizontalmente.



## Desenhar um percurso

Vamos agora desenhar o labirinto. Para tal, escolhemos o palco e clicamos no botão “pintar novo cenário” (fig. 16).



Figura 13. Alterar as propriedades de um ator



Figura 14. Alterar as propriedades de um ator

O palco onde decorre a ação vai poder assumir diferentes imagens ao longo de um projeto Scratch. Designaremos esses aspetos por “cenários”.



Figura 15. Alterar as propriedades de um ator

Figura 16. Desenhar um cenário

No lado direito do ecrã surge uma nova aba com um editor de imagem simples que permitirá desenhar o cenário do labirinto.

Na barra de ferramenta do lado esquerdo (fig. 17 A) escolhemos a ferramenta para traçar linhas. Se premirmos a tecla “shift” em simultâneo (seta utilizada para obter maiúsculas) teremos linhas apenas na horizontal ou vertical. Podemos ainda pintar – “encher” – o palco com uma cor, utilizando a ferramenta que está representada por um pequeno balde (fig. 17 A).

Vamos agora arrastar o nosso herói para a posição de partida. Se o tamanho do fantasma não estiver adequado ao percurso que desenhamos podemos utilizar o botão assinalado na figura 19 (A) para diminuir ou o comando ao lado para o aumentar.

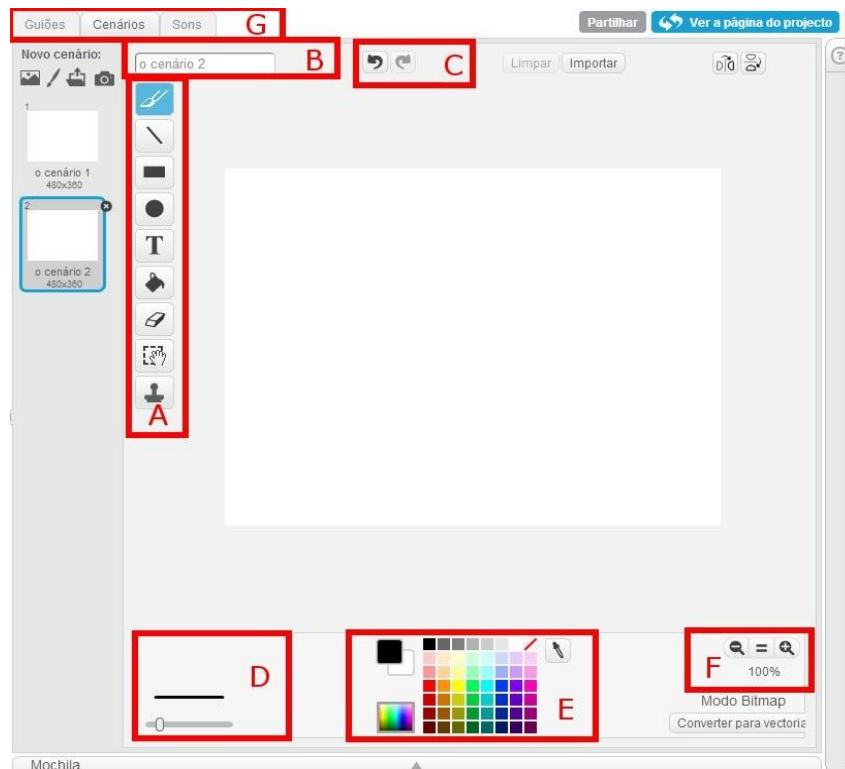


Figura 17. Alterar o cenário

### **Sempre que toca no labirinto, volta ao início**

Vamos agora programar o fantasma para que inicie o seu percurso na mesma posição e volte à esta posição sempre que tocar nas linhas do labirinto.

Um programa/jogo Scratch inicia, normalmente, quando carregamos na bandeira verde (Fig. 19 B), pelo que vamos começar um novo bloco de comandos com um bloco inicial que indicará os comandos a serem executados sempre que se carrega na bandeira verde para dar início ao jogo. Assim, arrastamos esse bloco (conjunto eventos) e em seguida o bloco “vai para as coordenadas...” presente no conjunto de bloco “movimento”. Repare que as coordenadas presentes neste bloco (fig. 20 A) correspondem à posição atual do fantasma que podemos confirmar nas suas propriedades (fig. 20 B).

Seguidamente, tente arrastar o fantasma para qualquer outra posição e iniciar o programa, clicando na bandeira verde. Se tudo estiver a funcionar, o fantasma deve posicionar-se na sua posição inicial sempre que a bandeira é premida. No entanto, se utilizarmos as setas para o movimentar e tocar no labirinto nada acontece.



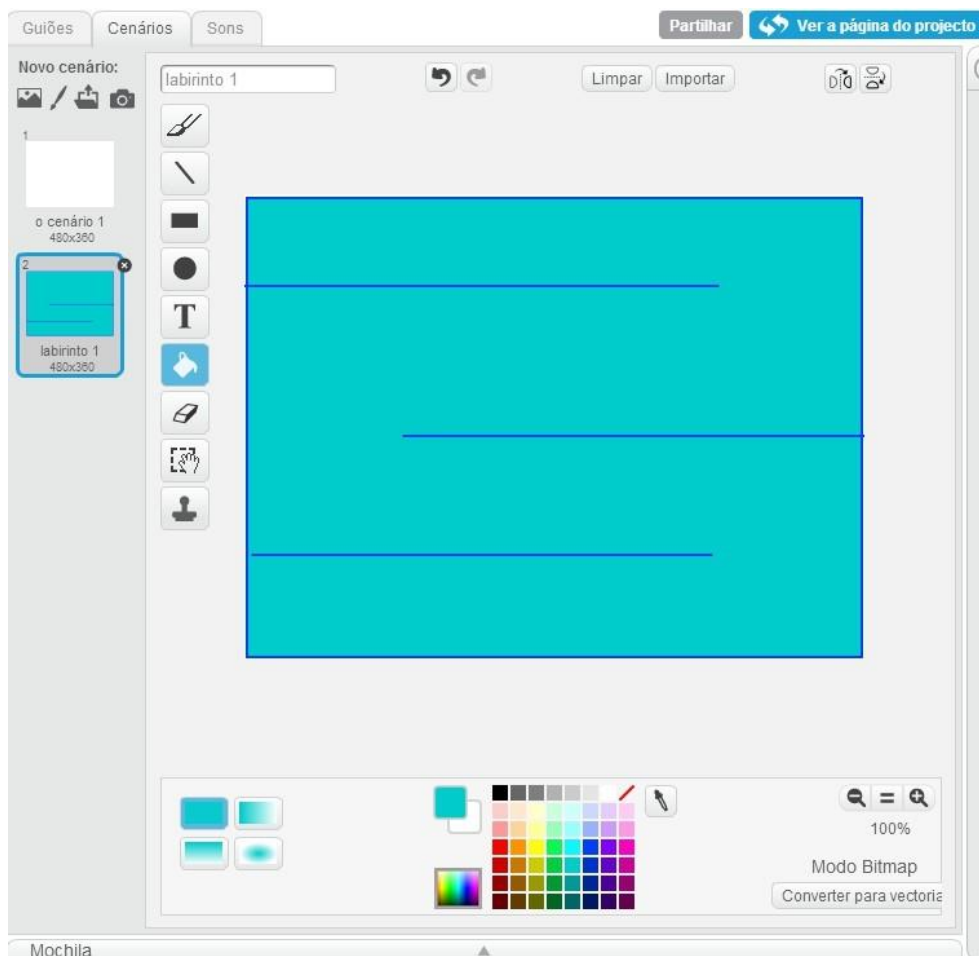


Figura 18. Desenhar um cenário

Para saber se toca no labirinto teremos que fazer constantemente um teste. Para isso vamos utilizar um ciclo “Para sempre” que arrastamos de “controlo” e dentro deste colocamos a condição “Se ...Então”.

Repare que temos que colocar uma condição no comando “Se ...Então”. Vamos procurar essa condição no conjunto de blocos “sensores” (Fig. 22). Arrastamos para o local da condição do comando “Se ...Então”. Em seguida escolhemos a cor, fazendo clique na cor presente no comando e depois na cor com que desenhemos o nosso labirinto.

Sempre que o fantasma tocar na cor com que desenhemos o labirinto deverá regressar ao ponto de partida. Assim, vamos arrastar para dentro do comando “Se ...Então” o comando que o reposiciona nesse lugar.

Podemos agora testar o nosso jogo e, se tudo estiver correto, o nosso fantasma deverá regressar à posição inicial sempre que toca no labirinto!

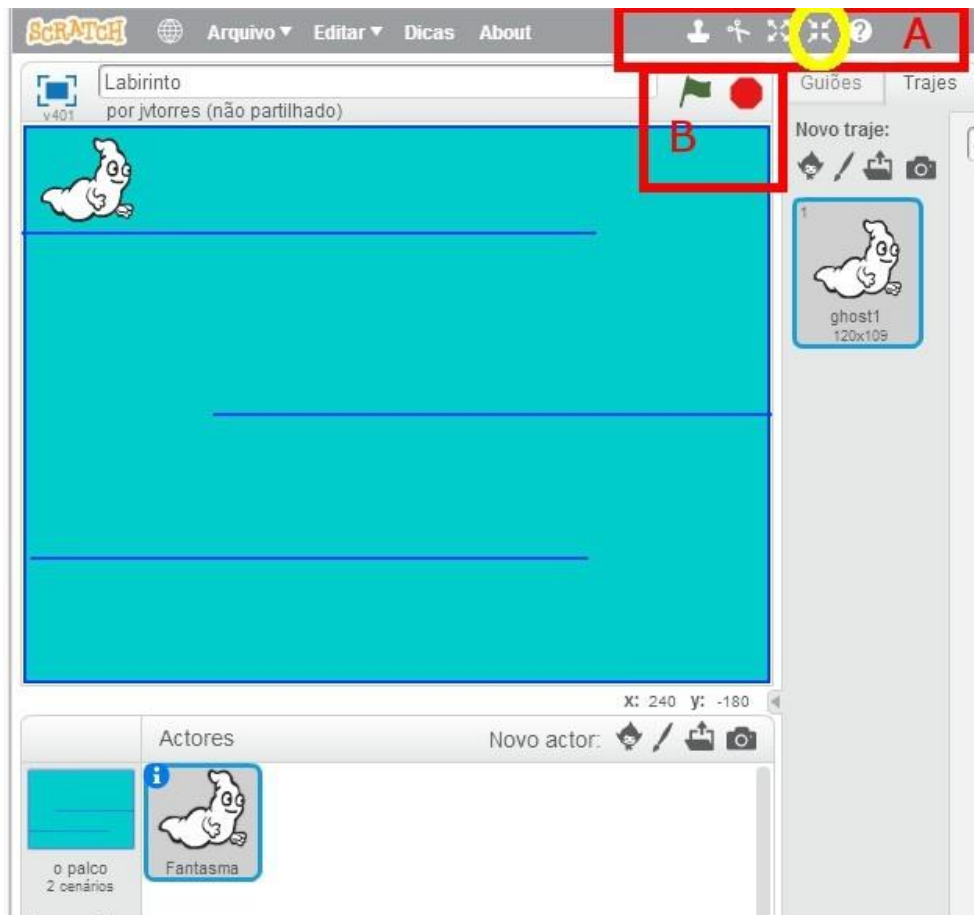


Figura 19. Posicionar o fantasma e alterar o seu tamanho

## Sons

Associado a cada um dos atores, temos “Guioes”, “Trajes” e “Sons”. Em “Guioes” definimos os comportamentos, o que devem fazer e quando. E “Trajes” o aspeto que podem tomar e em “Sons” vamos definir o som que podem produzir. Se passarmos para a aba referente ao som (Fig. 25 A), veremos que associado ao fantasma ha ja um som. Podemos tambem gravar novos sons ou importar sons gravados em ficheiro (fig. 25 B). Estes sons, associados as personagens ou cenrios, podem ser tocados, evocando-os a partir dos comandos associados no grupo “Som”. Assim, podemos juntar na nossa programao o comando que reproduz um pequeno som sempre que o fantasma chocar com o labirinto (fig. 24).

O nosso jogo deve funcionar, mas no temos ainda um objetivo final definido. Vamos precisar de outro ator que colocaremos no final do percurso. Sempre que o fantasma tocar nesse novo “personagem” exclamara “Conseguir!”

Vamos portanto a biblioteca de imagens (Fig. 6 A) e escolhemos uma nova personagem. No meu caso escolhi um presente (Gift). Arrastamos o presente para o final do percurso e escolhemos o tamanho que melhor se adequa (Fig. 27).

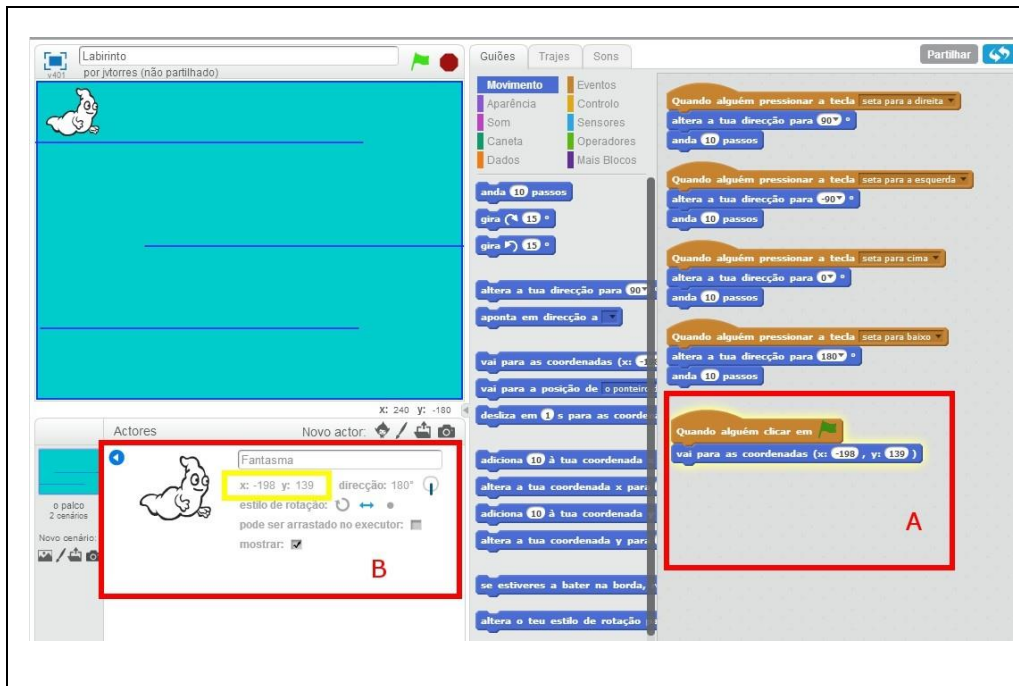


Figura 20. Bloco de comandos a executar no início



Figura 21. Se tocar então...

Pode agora importar outro som e fazer com que seja tocado sempre que o fantasma chegue ao fim do percurso!

### Guardar o trabalho

Estamos a trabalhar *on-line*, podemos guardar o nosso trabalho de duas formas: (i) transferindo o ficheiro para o nosso computador ou (ii) gravando no próprio site associando à nossa conta de utilizador. Para gravar o ficheiro no seu computador ou *pen*, aceda ao menu “Arquivo” e aí a “Descarregar para o meu computador”.



Figura 22. Se tocar na cor...

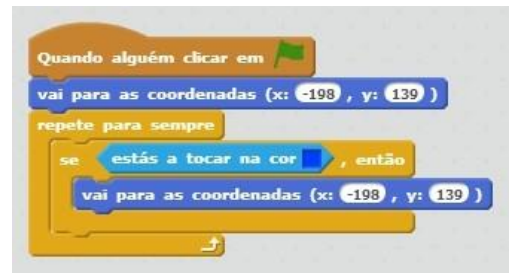


Figura 23. Programar tocar na cor



Figura 24. Tocar um som

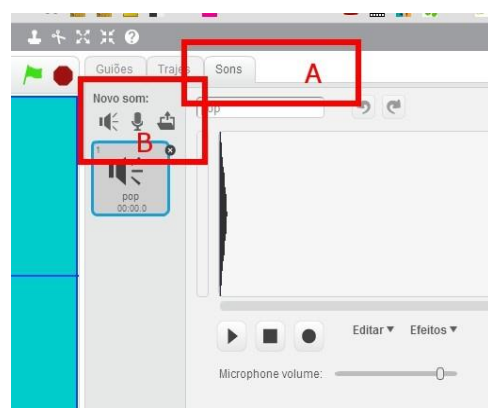


Figura 25. Sons associados às personagens

No entanto, é aconselhável criar uma conta no site acedendo no canto superior direito a autenticar e depois “aderir ao scratch”. Com uma conta poderá gravar automaticamente os seus trabalhos e partilhá-los. Poderá também fazer comentários a trabalhos de outros utilizadores e ter acesso ao código de milhares de programas!

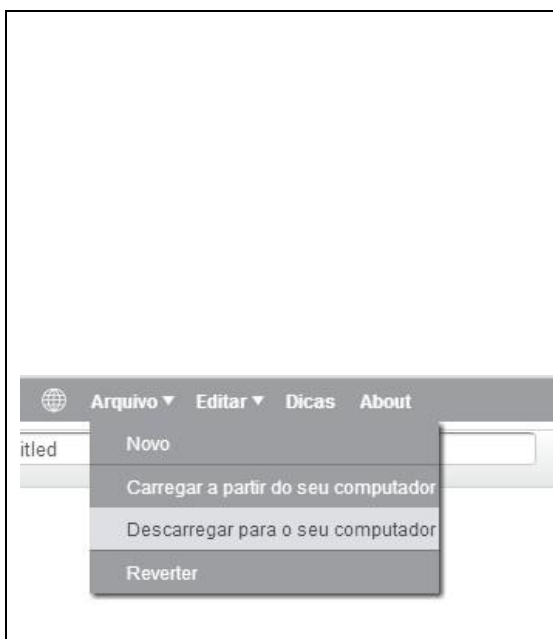


Figura 26. Descarregar para o computador

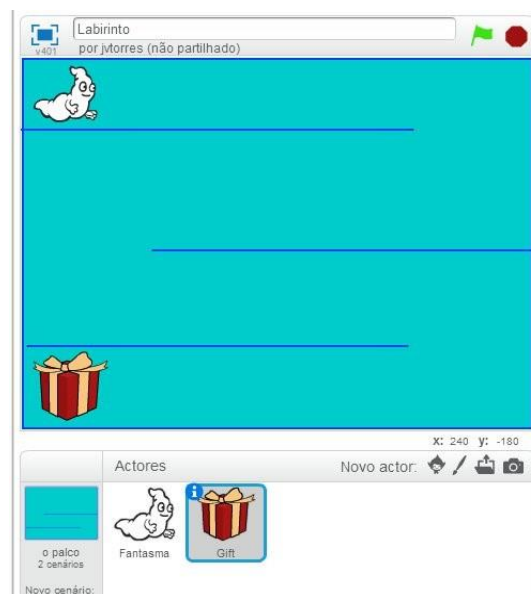


Figura 27 Um presente para objetivo final.

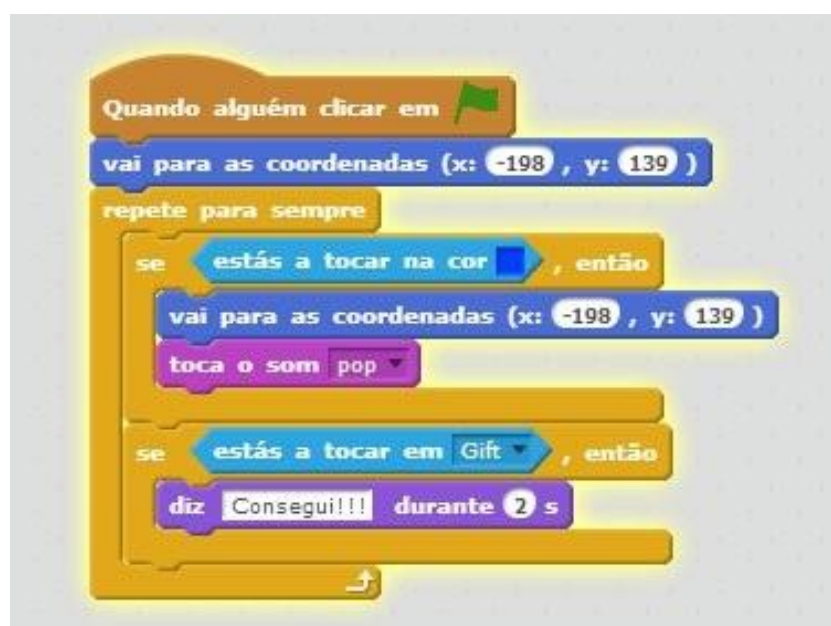


Figura 28. Parte do código final